#include <iostream>

#include <fstream>

using namespace std;

ifstream fin ("lista.in");

ofstream fout ("lista.out");

struct nod

{

int info;

nod \*urm;

};

nod \*prim, \*ultim, \*p, \*q;

int x, n;

void init (nod \*&prim, nod \*&ultim) { prim=ultim=NULL; }

int este\_vida(nod \*prim){return prim==NULL;}

void adaug\_nod(nod \*&prim, nod \*&ultim) {prim=new nod; prim->urm=NULL; ultim=prim;}

void adauga\_prim(nod \*&prim)

{

nod \*p=new nod;

p->info=x;

p->urm=prim;

prim=p;

}

void adauga\_ultim(nod \*&ultim)

{

nod \*p=new nod;

p->info=x;

p->urm=NULL;

ultim->urm=p;

ultim=p;

}

void adauga\_dupa(nod \*q, nod \*&ultim)

{

nod \*p=new nod;

p->info=x;

p->urm=q->urm;

q->urm=p;

if(q==ultim) ultim=p;

}

void adauga\_in\_fata(nod \*q, nod \*&ultim)

{

nod \*p=new nod;

p->info=q->info;

q->info=x;

p->urm=q->urm;

q->urm=p;

if(q==ultim) ultim=p;

}

void elimina\_prim(nod \*&prim) {nod \*q=prim; prim=prim->urm; delete q;}

void elimina\_ultim(nod \*prim, nod \*&ultim) { nod \*p, \*q=ultim;

for(p=prim; p->urm->urm!=NULL; p=p->urm); p->urm=NULL; ultim=p; delete q;}

void elimina(nod \*p, nod \*&ultim)

{

nod \*q=p->urm;

p->info=p->urm->info;

p->urm=p->urm->urm;

delete q;

if(p->urm==NULL) ultim=p;

}

void eliberare (nod \*&prim)

{

nod \* p=prim, \*q;

while(p!=NULL) {q=p; p=p->urm; delete q;}

prim=NULL;

}

int distincte (nod \*prim) {for(nod \*p=prim; p->urm!=NULL; p=p->urm)

for(nod \*q=p->urm; q!=NULL; q=q->urm)

if(p->info==q->info) return 0;

return 1; }

void prelucrare (nod \*p, nod \*&ultim)

{

for(nod \*p=prim; p!=NULL; p=p->urm)

if(p->info%2==1) adauga\_dupa(p, ultim);

}

void afisare(nod \*prim)

{

for(nod \*p=prim; p!=NULL; p=p->urm) fout<<p->info<<" ";

}

int main()

{

x=100;

fin>>n;

adaug\_nod(prim, ultim);

while(fin>>n) adauga\_ultim(ultim);

fin.close();

afisare(prim);

fout<<endl;

if (distincte(prim)) fout<<endl<<"lista are elem distincte"<<endl; else fout<<endl<<"lista nu are elem distincte"<<endl;

prelucrare(prim, ultim);

afisare(prim);

return 0;

}